```
DDDDDDDDDDD           CCCCCCCCCCCC   LLL
DDDDDDDDDDD           CCCCCCCCCCCC   LLL
DDDDDDDDDDD           CCCCCCCCCCCC   LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDD        DDD        CCC            LLL
DDDDDDDDDDD           CCCCCCCCCCCC   LLLLLLLLLLLLLL
DDDDDDDDDDD           CCCCCCCCCCCC   LLLLLLLLLLLLLL
DDDDDDDDDDD           CCCCCCCCCCCC   LLLLLLLLLLLLLLL
```

FILEID**EXIT

EXIT
LIS

EXIT
V04-000

J 16
- EXIT DCLS COMMAND EXECUTION   15-SEP-1984 23:45:56   VAX/VMS Macro V04-00   Page 1
                                        4-SEP-1984 23:40:28   [DCL.SRC]EXIT.MAR;1   (1)

```
0000     1              .TITLE  EXIT - EXIT DCLS COMMAND EXECUTION
0000     2              .IDENT  'V04-000'
0000     3
0000     4      ;
0000     5      ;**********************************************************************
0000     6      ;*                                                                    *
0000     7      ;*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                          *
0000     8      ;*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.           *
0000     9      ;*   ALL RIGHTS RESERVED.                                             *
0000    10      ;*                                                                    *
0000    11      ;*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000    12      ;*   ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE *
0000    13      ;*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER *
0000    14      ;*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000    15      ;*   OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY *
0000    16      ;*   TRANSFERRED.                                                      *
0000    17      ;*                                                                    *
0000    18      ;*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE *
0000    19      ;*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT *
0000    20      ;*   CORPORATION.                                                      *
0000    21      ;*                                                                    *
0000    22      ;*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS *
0000    23      ;*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.          *
0000    24      ;*                                                                    *
0000    25      ;*                                                                    *
0000    26      ;**********************************************************************
0000    27      ;
0000    28      ; W. H. BROWN 25-JUN-1977
0000    29      ;
0000    30      ; EXIT COMMAND EXECUTION
0000    31      ;
0000    32      ; MODIFIED BY:
0000    33      ;
0000    34      ;       V03-006 HWS0076         Harold Schultz  29-Jun-1984
0000    35      ;               Always run down images when EXIT command executed,
0000    36      ;               regardless of the indirect level.
0000    37      ;
0000    38      ;       V03-005 HWS0038         Harold Schultz  26-Mar-1984
0000    39      ;               Correct error message output in DCL$WAIT.
0000    40      ;
0000    41      ;       V03-004 PCG0006         Peter George    29-Jun-1983
0000    42      ;               Add home brew $SETIMR interlock mechanism.
0000    43      ;
0000    44      ;       V03-003 PCG0005         Peter George    03-May-1983
0000    45      ;               Specify REQIDT in SETIMR and CANTIM.
0000    46      ;
0000    47      ;       V03-002 PCG0004         Peter George    17-Jan-1983
0000    48      ;               Preserve status across subroutine calls.
0000    49      ;
0000    50      ;       V03-001 PCG0003         Peter George    30-Dec-1982
0000    51      ;               Set status on batch exit.
0000    52      ;---
```

```
         0000    54 ;
         0000    55 ; MACRO LIBRARY CALLS
         0000    56 ;
         0000    57
         0000    58         PRCDEF                          ; DEFINE PROCESS WORK AREA
         0000    59         WRKDEF                          ; DEFINE COMMAND WORK AREA
         0000    60         $CLIMSGDEF                      ; DEFINE CLI RELATED ERROR MESSAGES
         0000    61         $STSDEF                         ; DEFINE STATUS CODE FORMAT
         0000    62
     00000000    63         .PSECT  DCL$ZCODE,BYTE,RD,NOWRT
```

```
                              0000     65              .SBTTL  EXIT COMMAND
                              0000     66      ;+
                              0000     67      ; DCL$EXIT - EXIT COMMAND
                              0000     68      ;
                              0000     69      ; THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE EXIT DCLS COMMAND.
                              0000     70      ;
                              0000     71      ; INPUTS:
                              0000     72      ;
                              0000     73      ;       R10 = BASE ADDRESS OF COMMAND WORK AREA.
                              0000     74      ;       R11 = BASE ADDRESS OF PROCESS WORK AREA.
                              0000     75      ;
                              0000     76      ; OUTPUTS:
                              0000     77      ;
                              0000     78      ;       THE EXIT EXPRESSION IS EVALUATED
                              0000     79      ;
                              0000     80      ;       RO LOW BIT CLEAR INDICATES EXPRESSION EVALUATION FAILURE.
                              0000     81      ;
                              0000     82      ;             RO = DCL$_COMPLX - EXPRESSION TOO COMPLEX.
                              0000     83      ;             RO = DCL$_EXPSYN - EXPRESSION SYNTAX ERROR.
                              0000     84      ;             RO = DCL$_IVCHAR - INVALID CHARACTER IN NUMERIC STRING.
                              0000     85      ;             RO = DCL$_IVOPER - INVALID EXPRESSION OPERATOR.
                              0000     86      ;             RO = DCL$_UNDSYM - UNDEFINED SYMBOL.
                              0000     87      ;
                              0000     88      ;       IF THE EXPRESSION EVALUTES OK, THE INDIRECT LEVEL IS UNSTACKED
                              0000     89      ;       ONE LEVEL AND THE STATUS IS SET TO THE VALUE OF THE EXPRESSION,
                              0000     90      ;       OR IS LEFT AS THE ORIGINAL VALUE IF THE EXPRESSION WAS NULL.
                              0000     91      ;       RO LOW BIT SET INDICATES SUCCESSFUL COMPLETION.
                              0000     92      ;
                              0000     93      ;             RO = DCL$_NORMAL - NORMAL COMPLETION.
                              0000     94      ;-
                              0000     95
                              0000     96  DCL$EXIT::                               ; EXIT COMMAND
        10000000 8F C9       0000     97              BISL3   #STS$M_INHIB_MSG,-    ; GET LAST STATUS WITH INHIBIT MSG
        58    00B0 CB        0006     98                      PRC_L_LSTSTATUS(R11),R8 ;
              FFF3' 30       000A     99              BSBW    DCL$SETNBLK          ; FIND NON-BLANK CHARACTER
                 09 13       000D    100              BEQL    10$                  ; BR IF NONE-STATUS GETS SAVED
              FFEE' 30       000F    101              BSBW    DCL$BINEXPR          ; EVALUATE BINARY EXPRESSION
           1C 50 E9          0012    102              BLBC    RO,40$               ; BR IF ERROR EVALUATING EXPRESSION
           58    51 DO       0015    103              MOVL    R1,R8                ; SET RESULT OF EXPRESSION EVALUATION
      00B0 CB    58 DO       0018    104  10$:        MOVL    R8,PRC_L_LSTSTATUS(R11) ; STORE STATUS
              FFE0' 30       001D    105              BSBW    DCL$RUNDONI          ; RUN DOWN IMAGE BUT PRESERVE IND. LEVEL
              5C AB D5       0020    106              TSTL    PRC_L_INDEPTH(R11)   ; INDIRECT LEVEL 0?
                 OD 13       0023    107              BEQL    50$                  ; BR IF THERE IS NOTHING TO UNSTACK
              FFD8' 30       0025    108              BSBW    DCL$UNSTACK          ; REMOVE ONE LEVEL OF INDIRECT
        50    00B0 CB DO     0028    109  20$:        MOVL    PRC_L_LSTSTATUS(R11),RO ; SET RESULTANT STATUS
                 02 A8       002D    110              BISW    #WRK_M_COMMAND,-     ; SET COMMAND IN EXECUTION
              F0 AA          002F    111                      WRK_Q_FLAGS(R10)
                 05          0031    112  40$:        RSB                          ;
                             0032    113
                             0032    114      ;
                             0032    115      ; AN EXIT COMMAND ISSUED FOR COMMAND LEVEL 0
                             0032    116      ; THIS IS A CALL EXIT HANDLER/FORCE EXIT FOR INTERACTIVE JOBS AND
                             0032    117      ; A LOGOUT FOR BATCH JOBS.
                             0032    118      ;
              06 E1          0032    119  50$:        BBC     #PRC_V_MODE,-        ; BRANCH IF INTERACTIVE(IMAGE ALREADY
           F1 68 AB          0034    120                      PRC_Q_FLAGS(R11),20$ ;        RUN DOWN)
                             0037    121
```

```
 50    00B0 CB  D0  0037  122        MOVL    PRC_L_LSTSTATUS(R11),R0  ; SET STATUS
        FFC1'   30  003C  123        BSBW    DCL$SAVE_STATUS          ; SET $STATUS AND $SEVERITY
        FFBE'   31  003F  124        BRW     DCL$ABORT                ; FORCE LOGOUT
```

```
                              0042      126                .SBTTL  WAIT COMMAND
                              0042      127  ;+
                              0042      128  ; DCL$WAIT - WAIT COMMAND
                              0042      129  ;
                              0042      130  ; THIS ROUTINE IS CALLED AS AN INTERNAL COMMAND TO EXECUTE THE WAIT COMMAND.
                              0042      131  ;
                              0042      132  ; INPUTS:
                              0042      133  ;
                              0042      134  ;        R10 = BASE ADDRESS OF COMMAND WORK AREA.
                              0042      135  ;        R11 = BASE ADDRESS OF PROCESS WORK AREA.
                              0042      136  ;
                              0042      137  ;        THE NEXT TOKEN ON THE COMMAND LINE MUST BE A VALID
                              0042      138  ;        EXPRESSION FOR DELTA TIME MINUS THE NUMBER OF DAYS FIELD.
                              0042      139  ;
                              0042      140  ; OUTPUTS:
                              0042      141  ;
                              0042      142  ;        THE COMMAND FILE EXECUTION IS WAITED FOR THE SPECIFIED TIME.
                              0042      143  ;-
                              0042      144  DCL$WAIT::
              FFBB'   30      0042      145                BSBW      DCL$MARK                    ; MARK CURRENT POSITION IN BUFFER
      52   F486 CA    D0      0045      146                MOVL      WRK_L_EXPANDPTR(R10),R2     ; COPY START OF TIME FIELD
      62   2030 8F    B0      004A      147                MOVW      #^A70-/,(R2)                ; APPEND ZERO DAY COUNT FOR WAIT
   F486 CA     02     C0      004F      148                ADDL      #2,WRK_L_EXPANDPTR(R10)     ; UPDATE EXPANSION POINTER
              FFA9'   30      0054      149  10$:          BSBW      DCL$MOVCHAR                 ; MOVE A CHARACTER TO PARSE BUFFER
                FB    12      0057      150                BNEQ      10$                         ; UNTIL END OF LINE
              FFA4'   30      0059      151                BSBW      DCL$MARKEDTOKEN             ; GET DESCRIPTOR OF TIME STRING
                51    D7      005C      152                DECL      R1                          ; DISCOUNT EOL CHARACTER
                06    BB      005E      153                PUSHR     #^M<R1,R2>                  ; BUILD DESCRIPTOR FOR TIME STRING
      57     66 AB    3E      0060      154                MOVAW     PRC_W_WAITIOSB(R11),R7      ; GET WAIT IOSB ADDRESS
                              0064      155                $CANTIM_S REQIDT=R7                   ; CANCEL ANY PREVIOUS TIMERS
      53     5E      D0       006F      156                MOVL      SP,R3                       ; SAVE ADDRESS OF QUAD WORD DESCRIPTOR
                              0072      157                $BINTIM_S (R3),(R3)                   ; CONVERT TIME TO BINARY DELTA FORMAT
             2D 50    E9      007D      158                BLBC      R0,30$                      ; BR IF ERROR IN TIME
                67    B4      0080      159                CLRW      (R7)                        ; CLEAR WAIT IOSB STATUS
                              0082      160                $SETIMR_S EFN=#EXE$C_SYSEFN,-         ; SET TIMER FOR SPECIFIED TIME
                              0082      161                          DAYTIM=(R3),-
                              0082      162                          ASTADR=WAIT_AST,-
                              0082      163                          REQIDT=R7
             11 50    E9      0099      164                BLBC      R0,30$                      ; BR IF ERROR SETTING TIMER
                              009C      165                $SYNCH_S  IOSB=(R7),-                 ; WAIT THE SPECIFIED TIME
                              009C      166                          EFN=#EXE$C_SYSEFN
                07    11      00AB      167                BRB       35$                         ; GOTO EXIT
                              00AD      168  30$:          STATUS    IVDTIME                     ; SET INVALID DELTA TIME.
                8E    7C      00B4      169  35$:          CLRQ      (SP)+                       ; REMOVE SCRATCH QUAD WORD FROM STACK
                05            00B6      170                RSB
```

EXIT
V04-000
    - EXIT DCLS COMMAND EXECUTION   C 1   15-SEP-1984 23:45:56 VAX/VMS Macro V04-00   Page 6
    WAIT AST ROUTINE   4-SEP-1984 23:40:28 [DCL.SRC]EXIT.MAR;1   (5)

```
                        00B7    172                 .SBTTL   WAIT AST ROUTINE
                        00B7    173   ;+
                        00B7    174   ; WAIT_AST - WAIT AST ROUTINE
                        00B7    175   ;
                        00B7    176   ; THIS ROUTINE PROVIDES THE WAIT EVENT FLAG SYNCHRONIZATION.
                        00B7    177   ;
                        00B7    178   ; INPUTS:
                        00B7    179   ;
                        00B7    180   ;        NONE
                        00B7    181   ;
                        00B7    182   ; OUTPUTS:
                        00B7    183   ;
                        00B7    184   ;        R0 = SUCCESS
                        00B7    185   ;-
                        00B7    186
                        00B7    187   WAIT_AST:
                0080    00B7    188           .WORD    ^M<R7>                  ;FLUSH OUTPUT
  04 BC   01    B0      00B9    189           MOVW     #1,@4(AP)               ;SET THE STATUS IN THE WAIT IOSB
                        00BD    190           STATUS   NORMAL                  ;SET SUCCESSFUL STATUS
                04      00C4    191           RET                             ;
                        00C5    192
                        00C5    193           .END
```

Symbol	Value			Symbol	Value		
$$T1	= 00000000			PRC_L_STACKLM	000000A4		
CLI$_IVDTIME	= 00038298			PRC_L_STACKPT	000000A0		
CLI$_NORMAL	= 00030001			PRC_L_STATUS	00000054		
DCL$ABORT	********	X	02	PRC_L_STS	00000084		
DCL$BINEXPR	********	X	02	PRC_L_STV	00000088		
DCL$EXIT	00000000	RG	02	PRC_L_SYMBOL	00000060		
DCL$MARK	********	X	02	PRC_L_TMBX	00000074		
DCL$MARKEDTOKEN	********	X	02	PRC_L_TRMLIST	00000010		
DCL$MOVCHAR	********	X	02	PRC_Q_ALLOCREG	00000020		
DCL$RUNDWNI	********	X	02	PRC_Q_COMMAND	000000E0		
DCL$SAVE_STATUS	********	X	02	PRC_Q_FLUSHTIME	000000D0		
DCL$SETNBLK	********	X	02	PRC_Q_GLOBAL	00000028		
DCL$UNSTACK	********	X	02	PRC_Q_IMAGENAME	000000D8		
DCL$WAIT	00000042	RG	02	PRC_Q_KEYPAD	00000040		
EXE$C_SYSEFN	********	X	02	PRC_Q_LABEL	00000030		
PRC_B_CONTINUE	000000F3			PRC_Q_LOCAL	00000038		
PRC_B_DEFRADIX	000000AE			PRC_Q_SAVEPRIV	000000E8		
PRC_B_EXMDEPMOD	000000AD			PRC_T_OUTDVI	0000011C		
PRC_B_EXMDEPWID	000000AC			PRC_V_MODE	= 00000006		
PRC_B_EXONLYL	0000012D			PRC_W_ASTIOSB	0C0000C6		
PRC_B_FLAGS2	000000AF			PRC_W_ASTRETN	000000C8		
PRC_B_IMGFLAG	00000078			PRC_W_ASTSTATUS	000000C4		
PRC_B_OUTFLAGS	0000012C			PRC_W_ATTMBX	0000007A		
PRC_B_PROMPTLEN	000000F0			PRC_W_FLAGS	00000068		
PRC_C_LENGTH	00000534			PRC_W_INPCHAN	00000064		
PRC_G_COMMANDS	00000133			PRC_W_ONLEVEL	0000006A		
PRC_G_PROMPT	000000F4			PRC_W_OUTIFI	00000114		
PRC_K_LENGTH	00000534			PRC_W_OUTISI	0C000116		
PRC_L_CURRKEY	00000048			PRC_W_OUTMBXCHN	000000CA		
PRC_L_EXMDEPADR	000000A8			PRC_W_OUTMBXREF	000000CE		
PRC_L_EXTARG	00000094			PRC_W_OUTMBXSIZ	000000CC		
PRC_L_EXTBLK	0000008C			PRC_W_PMPTCTRL	000000F1		
PRC_L_EXTCOD	0000009C			PRC_W_WAITIOSB	00000066		
PRC_L_EXTHND	00000090			STS$M_INHIB_MSG	= 10000000		
PRC_L_EXTPRM	00000098			SYS$BINTIM	********	GX	02
PRC_L_IDFLNK	00C000BC			SYS$CANTIM	********	GX	02
PRC_L_IMGACTSTS	00000080			SYS$SETIMR	********	GX	02
PRC_L_INDCLOCK	0000007C			SYS$SYNCH	********	GX	02
PRC_L_INDEPTH	0000005C			WAIT_AST	000000B7	R	02
PRC_L_INDFAB	0000001C			WRK_B_CMDOPT	FFFFFFC3		
PRC_L_INDINPRAB	00000014			WRK_B_MAXPARM	FFFFFFD0		
PRC_L_INDOUTRAB	00000018			WRK_B_MINPARM	FFFFFFD1		
PRC_L_INPRAB	00000008			WRK_B_PARMCNT	FFFFFFCE		
PRC_L_LASTKEY	0000004C			WRK_B_PARMSUM	FFFFFFCF		
PRC_L_LSTSTATUS	000000B0			WRK_B_RECALLCNT	FFFFFFC5		
PRC_L_ONCTLY	000000B8			WRK_B_VALLEV	FFFFFFC4		
PRC_L_ONERROR	0000006C			WRK_B_VERBTYP	FFFFFFC2		
PRC_L_OUTOFBAND	000000B4			WRK_C_LENGTH	FFFFF486		
PRC_L_OUTRAB	0000000C			WRK_G_BUFFER	FFFFF492		
PRC_L_OUTRABCTX	00000118			WRK_G_INPBUF	FFFFF896		
PRC_L_PPFLIST	00000070			WRK_G_RESULT	FFFFF9B6		
PRC_L_RECALLPTR	0000012F			WRK_K_LENGTH	FFFFF486		
PRC_L_RESTART	00000058			WRK_L_CHARPTR	FFFFF48E		
PRC_L_SAVAP	00000000			WRK_L_DISALLOW	FFFFFFE6		
PRC_L_SAVFP	00000004			WRK_L_ERRORRTN	FFFFF9AE		
PRC_L_SEVERITY	00000050			WRK_L_EXPANDPTR	FFFFF486		
PRC_L_SPWN	000000C0			WRK_L_IMAGE	FFFFFFE2		

```
WRK_L_MARKPTR          FFFFF48A
WRK_L_PAROUT           FFFFFFD2
WRK_L_PMPTADDR         FFFFF9A2
WRK_L_PROMPTRTN        FFFFF9A6
WRK_L_PROPTR           FFFFFFC6
WRK_L_QUABLK           FFFFFFCA
WRK_L_READRTN          FFFFF9AA
WRK_L_RECALLPTR        FFFFFFEA
WRK_L_RSLEND           FFFFFFB6
WRK_L_RSLNXT           FFFFFFBA
WRK_L_SAVAP            FFFFFFF8
WRK_L_SAVFP            FFFFFFFC
WRK_L_SAVSP            FFFFFFF4
WRK_L_SIGNALRTN        FFFFFFD6
WRK_L_SPECRTN          FFFFF9B2
WRK_L_TAB_VEC          FFFFFFDE
WRK_L_VERB             FFFFFFBE
WRK_M_COMMAND        = 00000002
WRK_W_FLAGS            FFFFFFF0
WRK_W_FLAGS2           FFFFFFF2
WRK_W_IMGCHAN          FFFFFFEE
WRK_W_PMPTLEN          FFFFF99E
```

```
+-----------------+
! Psect synopsis !
+-----------------+
```

PSECT name	Allocation		PSECT No.		Attributes											
. ABS .	00000000	(0.)	00 (	0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD		NOWRT	NOVEC	BYTE
ABS	FFFFFFFC	(0.)	01 (	1.)	NOPIC	USR	CON	ABS	LCL	NOSHR	EXE	RD		WRT	NOVEC	BYTE
DCL$ZCODE	000000C5	(197.)	02 (	2.)	NOPIC	USR	CON	REL	LCL	NOSHR	EXE	RD	NOWRT		NOVEC	BYTE

```
+-------------------------+
! Performance indicators !
+-------------------------+
```

Phase	Page faults	CPU Time	Elapsed Time
Initialization	14	00:00:00.06	00:00:01.78
Command processing	85	00:00:00.65	00:00:06.71
Pass 1	187	00:00:05.83	00:00:23.62
Symbol table sort	0	00:00:00.59	00:00:01.61
Pass 2	38	00:00:00.95	00:00:03.46
Symbol table output	16	00:00:00.10	00:00:00.12
Psect synopsis output	2	00:00:00.04	00:00:00.24
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	342	00:00:08.22	00:00:37.68

The working set limit was 1050 pages.
24786 bytes (49 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 440 non-local and 7 local symbols.
193 source lines were read in Pass 1, producing 13 object records in Pass 2.
35 pages of virtual memory were used to define 22 macros.

```
                                     +-----------------------------+
                                     ! Macro library statistics !
                                     +-----------------------------+

Macro library name                        Macros defined
------------------                        --------------
_$255$DUA28:[SYSLIB]SYSBLDMLB.MLB;1              0
_$255$DUA28:[DCL.OBJ]DCL.MLB;1                   5
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                   0
_$255$DUA28:[SYSLIB]STARLET.MLB;2               12
TOTALS (all libraries)                          17
```
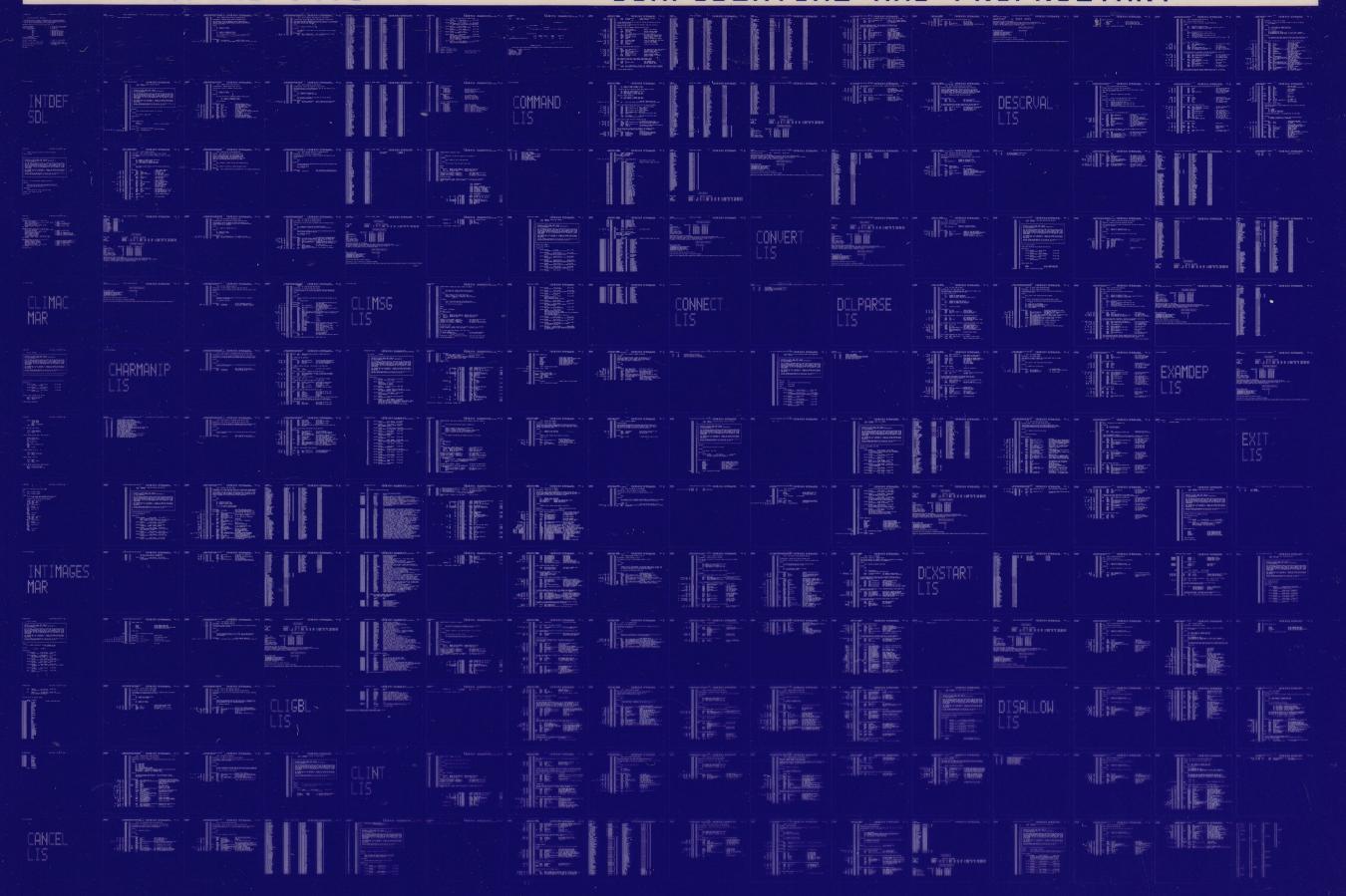
604 GETS were required to define 17 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:EXIT/OBJ=OBJ$:EXIT MSRC$:EXIT/UPDATE=(ENH$:EXIT)+EXECML$/LIB+LIB$:DCL/LIB+SYS$LIBRARY:SYSBLDMLB/LIB

INTDEF
SDL

COMMAND
LIS

DESCRVAL
LIS

CONVERT
LIS

CLIMAC
MAR

CLIMSG
LIS

CONNECT
LIS

DCLPARSE
LIS

CHARMANIP
LIS

EXAMDEP
LIS

EXIT
LIS

INTIMAGES
MAR

DCXSTART
LIS

CLIGBL
LIS

DISALLOW
LIS

CLINT
LIS

CANCEL
LIS